Use Case 3

Štefan Lyócsa

Masaryk University

Goal

The goal is to predict loan default directly (1 - loan defaulted, 0 - otherwise). If the return on the loan is negative, we consider the loan to be defaulted. For this purpose we use **logistic regression**. We will use factor network model to extract network variables, hoping to increase predictions of default.

- We create an adjacency matrix.
- We calculate vertex level network variables.
- We augment logistic regression models with new variables.
- We compare the forecasting accuracy.

Recall, that in logistic regression we model logarithm of odds, that we refer to as $logit(p_i)$:

$$log(ODDS_i) = log(\frac{p_i}{1-p_i}) = logit(p_i) = \beta_0 + \beta_1 \times Int_i$$

Coefficients are estimated as:

$$\underset{\hat{\beta}_0,\hat{\beta}_1}{max} \rightarrow LL(\beta_0, \beta_1) = \sum_{i=1}^{n} y_i log(p_i) + (1 - y_i) log(1 - p_i)$$

We estimate logistic regression on our P2P dataset - similarly as before. Load the dataset - again:

Number of predicted loans: Loans where we want to predict whether they default:

- `NF = 500`
- `N = dim(DT)[1]`

Storing the dependent variables for each of the samples:

- `DT$DEF = (DT$RR2<0)*1`

The sample to use to estimate the model:

- `S1 = DT[1:(N-NF),]`

The sample to use to predict (out-of-sample) loan return:

- `S2 = DT[(N-NF+1):N,]`

Estimate standard model:

- `m4 = glm(formula = DEF  new+ver3+ver4+lfi+lee+luk+lrs+ female+lamt+int+durm+educprim+educbasic+ educvocat+educse espem+esfue+essem+esent+esret+dures+exper+ linctot+noliab lamntplr+lamteprl+nopearlyrep, family = binomial(link = "logit"),data = S1)`
- `summary(m4)`

Predict defaults:

- `ypred = predict(m4,new=S2)`
- `ypred = exp(ypred)/(1+exp(ypred))`
- `ytrue = S2$DEF`

Evaluate default predictions

- Ratio of the number of **correctly predicted defaults** to the number of **true defaults**. True positive rate (TPR). Large TPR means we are good at predicting defaults.
- Ratio of the number of **wrongly predicted defaults** to the number of **true non-defaults**. False positive rate (FPR). Low FPR means we are good and predicting non-defaults.

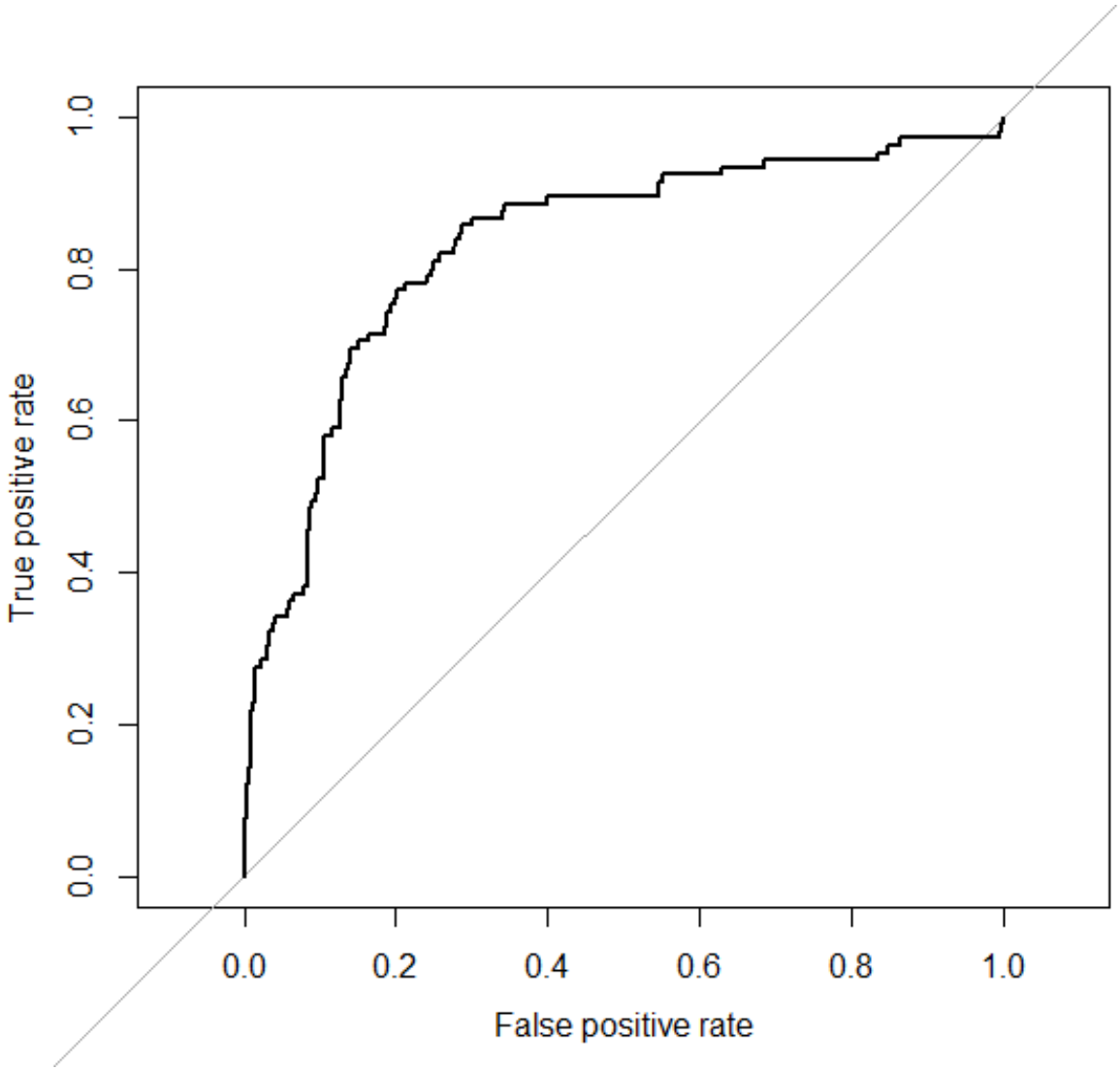
The higher the TPR and the lower the FPR the better, thus we wish to maximize the ratio of TPR/FPR. Plotting TPR (y-axis) against FPR (x-axis) leads to the **receiver operating characteristic curve**, the so called ROC.

Evaluate default predictions

- `library(pROC)`
- `roc_obj <- roc(ytrue, ypred)`
- `plot(roc_obj, xlab = "False positive rate", ylab = "True positive rate")`

Area under the curve (AUC):

- `LOGIT = auc(roc_obj)`


True positive rate
0.0
0.2
0.4
0.6
0.8
1.0
False positive rate
0.0
0.2
0.4
0.6
0.8
1.0

In standard logistic model, the coefficients are estimated as:

$$\underset{\hat{\beta}_0,\hat{\beta}_1}{max} \rightarrow \sum_{i=1}^{n} y_i log(p_i) + (1 - y_i) log(1 - p_i)$$

In LASSO logistic model, the coefficients are estimated as:

$$\underset{\hat{\beta}_0,\hat{\beta}_1}{max} \rightarrow \sum_{i=1}^{n} [y_i log(p_i) + (1 - y_i) log(1 - p_i)] - \lambda \sum_{j=1}^{p} |\beta_j|$$

Preparing data

Define the matrix of input and output variables:

- `indep = as.matrix(DT[1:(N-NF),c('new','ver3','ver4','lf`
`'undG','female','lamt','int','durm','educprim','educbasic`
`'educvocat','educsec','msmar','msco','mssi','msdi','nrode`
`'espem','esfue','essem','esent','esret','dures','exper',`
`'linctot','noliab','lliatot','norli','noplo','lamountplo'`
`'lamntplr','lamteprl','nopearlyrep','Deg','Hac',paste('g'`
- `dep = DT[1:(N-NF),'DEF']`

Preparing data

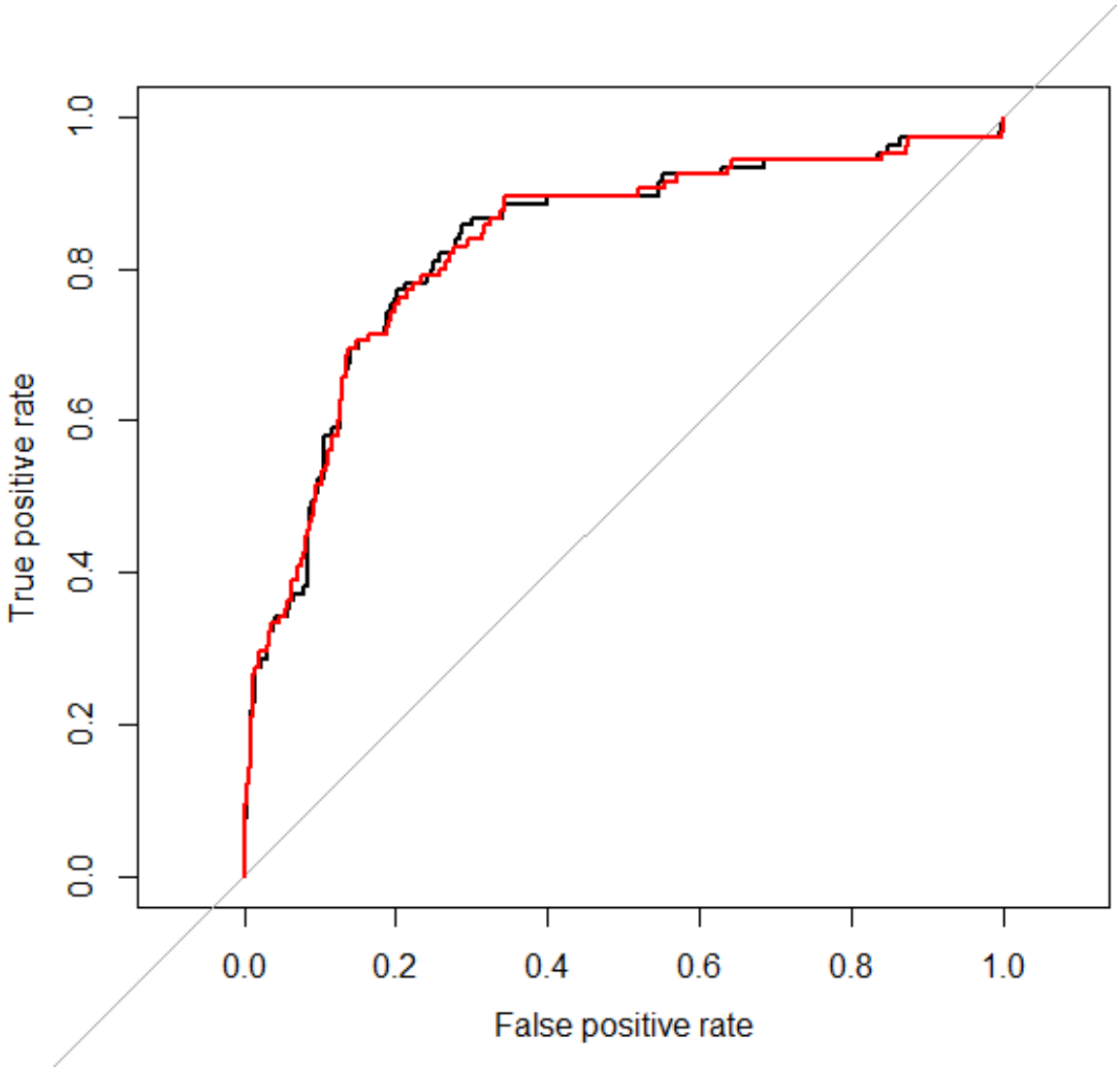
- `pred = as.matrix(DT[(N-NF+1):N,c('new','ver3','ver4','l`
`'undG','female','lamt','int','durm','educprim','educbasic`
`'educvocat','educsec','msmar','msco','mssi','msdi','nrode`
`'espem','esfue','essem','esent','esret','dures','exper',`
`'linctot','noliab','lliatot','norli','noplo','lamountplo'`
`'lamntplr','lamteprl','nopearlyrep','Deg','Hac',paste('g'`
- `ytrue = DT[(N-NF+1):N,'DEF']`

Estimating the model:

- `m5 = cv.glmnet(x=indep, y=dep, type.measure='auc',alpha`
- `coef(m5, s = "lambda.1se")`

Predicting defaults:

- `ypred = predict(m5,newx=pred,s=m5$lambda.1se)`
- `ypred = exp(ypred)/(1+exp(ypred))`
- `roc_obj <- roc(ytrue, ypred)`
- `lines(roc_obj, col='red')`
- `LOGIT_LASSO = auc(roc_obj)`


True positive rate
0.0
0.2
0.4
0.6
0.8
1.0
False positive rate
0.0
0.2
0.4
0.6
0.8
1.0

Create network in R

Import data again (but do not delete anything from previous Use Case). We arbitrarily select variables that we think might identify a bad loan:

- `X = DT[,c('int','durm','linctot','noliab')]`

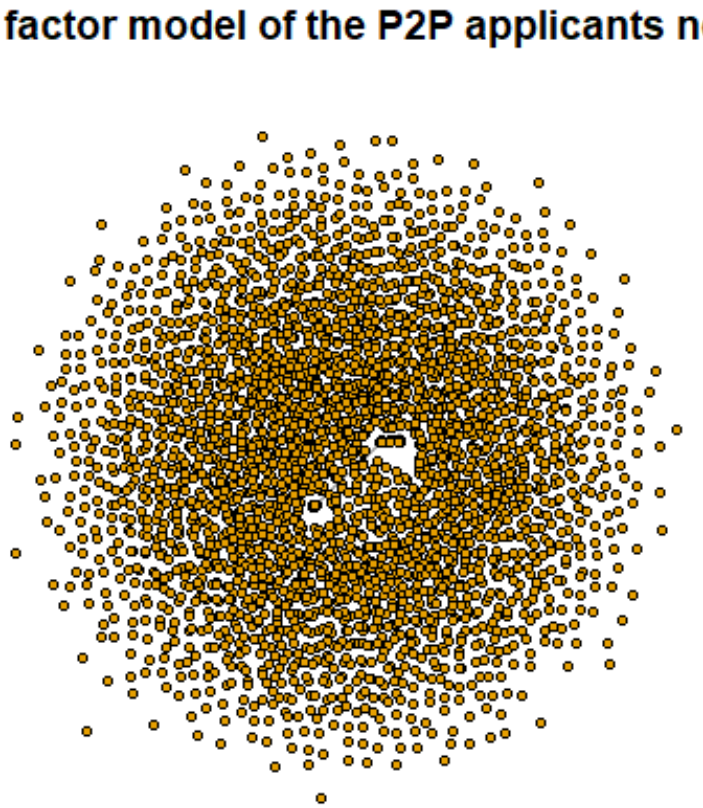
Run the following function:

- `AM = FN_SVD(X,p=0.75,gam=0.10)`
- `g = graph_from_adjacency_matrix(AM, mode = 'undirected', weighted = TRUE)`

We can visualize the Network Factor Model:

- `plot(g, graph = 'NFM', vertex.label=NA,vertex.size = 3, main = 'Network factor model of the P2P applicants networks')`

Create the network in R



- vertex degree,
- harmonic centrality,
- Community detection - Louvain method.

To address the issue of isolated vertices, one can assume that the shortest distance between vertex i and an isolated vertex j is ∞, while conveniently assuming that $1/\infty = 0$. Harmonic centrality is therefore:
$H_(i) = \sum_{d_(i,j)<\infty, i \neq j} \frac{1}{d_(i,j)}$
where $d_(i,j)$ is the shortest path from vertex i to vertex j in the network.

Estimating vertex level attributes in R

The following function calculates centrality and community:

- `NetDscr=BVC(g)`

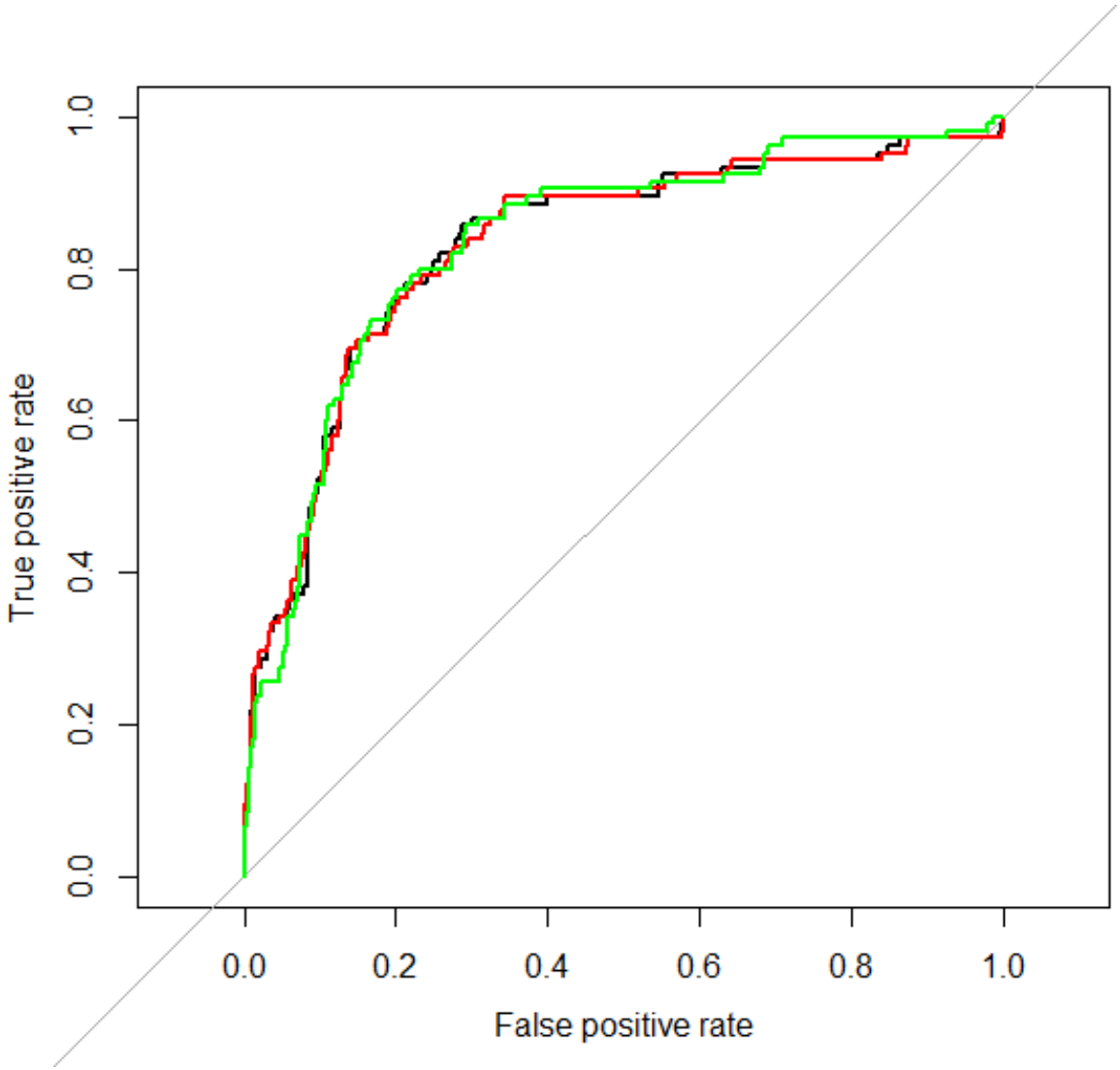
Now add variable into the model:

- `DT$Deg = NetDscr$VCentrality[,1]`
- `DT$Hac = NetDscr$VCentrality[,2]`
- `DT = data.frame(DT,NetDscr$Community)`

- `S1 = DT[1:(N-NF),]`
- `S2 = DT[(N-NF+1):N,]`
- `m6 = glm(formula = DEF  new+ver3+ver4+lfi+lee+luk+lrs+ female+lamt+int+durm+educprim+educbasic+ educvocat+educse espem+esfue+essem+esent+esret+dures+exper+ linctot+noliab lamntplr+lamteprl+nopearlyrep+Deg+Hac+g1+g2+g3+g4, family = binomial(link = "logit"),data = S1)`
- `summary(m6)`

Predicting defaults:

- `ypred = predict(m6,new=S2)`
- `ypred = exp(ypred)/(1+exp(ypred))`
- `ytrue = S2$DEF`
- `roc_obj <- roc(ytrue, ypred)`
- `lines(roc_obj,col='green')`
- `LOGIT_N = auc(roc_obj)`


True positive rate
0.0
0.2
0.4
0.6
0.8
1.0
False positive rate
0.0
0.2
0.4
0.6
0.8
1.0

Preparing data

Define the matrix of input and output variables:

- `indep = as.matrix(DT[1:(N-NF),c('new','ver3','ver4','lf`
`'undG','female','lamt','int','durm','educprim','educbasic`
`'educvocat','educsec','msmar','msco','mssi','msdi','nrode`
`'espem','esfue','essem','esent','esret','dures','exper',`
`'linctot','noliab','lliatot','norli','noplo','lamountplo'`
`'lamntplr','lamteprl','nopearlyrep','Deg','Hac',paste('g'`
- `dep = DT[1:(N-NF),'RR2']`

Preparing data

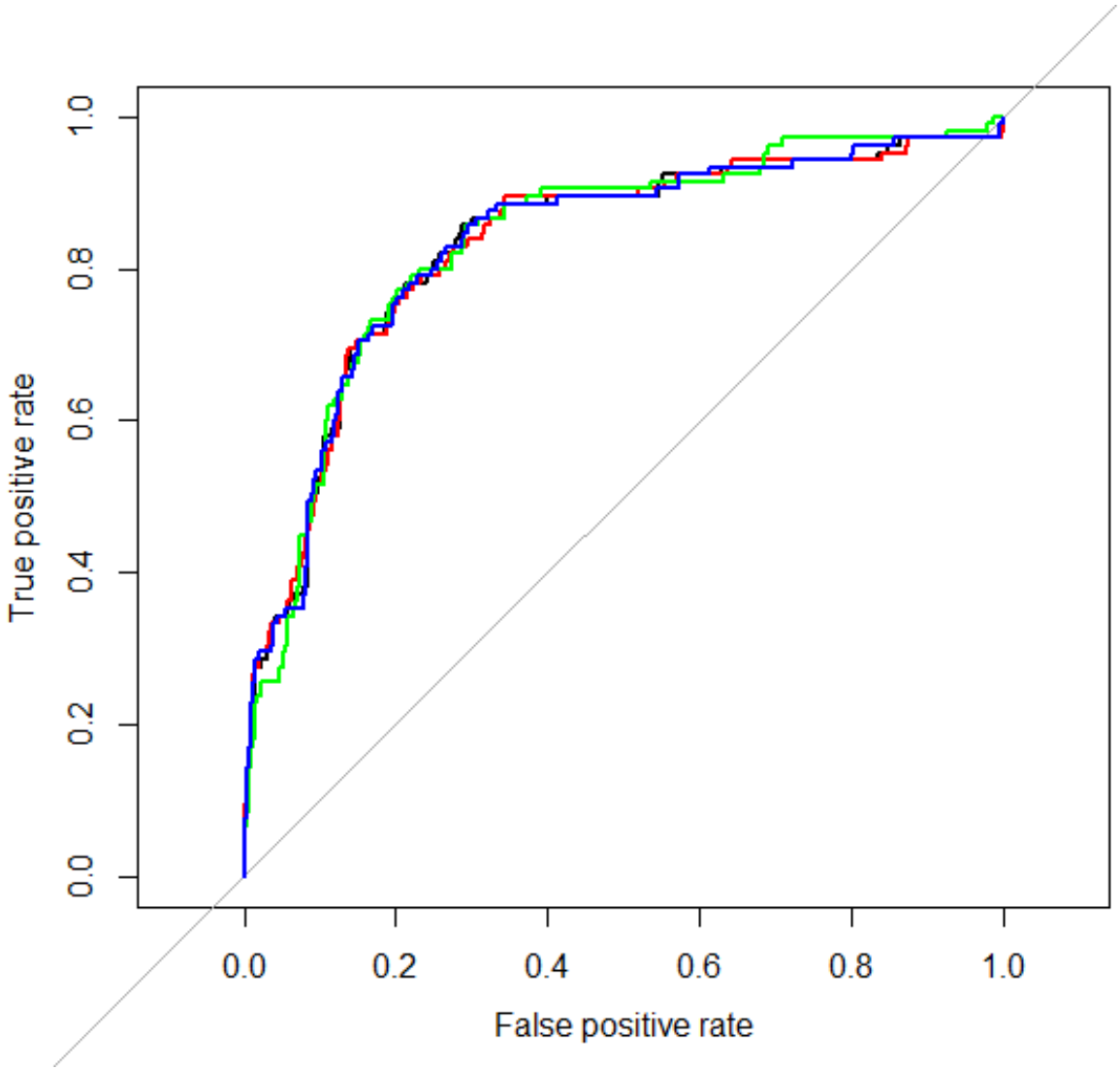
- `pred = as.matrix(DT[(N-NF+1):N,c('new','ver3','ver4','l`
`'undG','female','lamt','int','durm','educprim','educbasic`
`'educvocat','educsec','msmar','msco','mssi','msdi','nrode`
`'espem','esfue','essem','esent','esret','dures','exper',`
`'linctot','noliab','lliatot','norli','noplo','lamountplo'`
`'lamntplr','lamteprl','nopearlyrep','Deg','Hac',paste('g'`
- `ytrue = DT[(N-NF+1):N,'RR2']`

Estimating the model:

- `m7 = cv.glmnet(x=indep, y=dep, type.measure=äuc",alpha=`
- `coef(m7, s = "lambda.1se")`

Predicting defaults:

- `ypred = predict(m7,newx=pred,s=m7$lambda.1se)`
- `ypred = exp(ypred)/(1+exp(ypred))`
- `roc_obj <- roc(ytrue, ypred)`
- `lines(roc_obj,col='blue')`
- `LOGIT_N_LASSO = auc(roc_obj)`


True positive rate
0.0
0.2
0.4
0.6
0.8
1.0
False positive rate
0.0
0.2
0.4
0.6
0.8
1.0

- `AUC = c(LOGIT,LOGIT_LASSO,LOGIT_N,LOGIT_N_LASSO)`
- `names(AUC) = c("Logit","Logit-L","Logit-N","Logit-NL")`
- `AUC = sort(AUC,decreasing=T)`
- `cbind(AUC)`

	AUC
Logit-N	0.8338517
Logit	0.8326703
Logit-NL	0.8305726
Logit-L	0.8300904

Use Case 3

Štefan Lyócsa

Masaryk University